AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Original Claim) A method for generating code to perform anticipatory prefetching for data references, comprising:

receiving code to be executed on a computer system;

analyzing the code to identify data references to be prefetched; and

inserting prefetch instructions into the code in advance of the identified data references, wherein inserting the prefetch instructions involves,

attempting to calculate a stride value for a given data reference within a loop,

if the stride value cannot be calculated, setting the stride value to a default stride value, and

inserting a prefetch instruction to prefetch the given data reference for a subsequent loop iteration based on the stride value;

wherein the stride value is constant for some (but not necessarily all) loop iterations.

2. (Original Claim) The method of claim 1, further comprising allowing a system user to specify the default stride value.

3. (Original Claim) The method of claim 1, wherein calculating the stride value involves:

identifying an induction variable for the stride value;

identifying a stride function for the stride value; and

calculating the stride value based upon the stride function and the induction variable.

4. (Original Claim) The method of claim 1, wherein inserting the prefetch instruction based on the stride value involves:

calculating a prefetch cover distance by dividing a cache line size by the stride value;

calculating a prefetch ahead distance as a function of a prefetch latency, the prefetch cover distance and an execution time of a loop; and

calculating a prefetch address by multiplying the stride value by the prefetch cover distance and the prefetch ahead distance and adding the result to an address accessed by the given data reference.

5. (Original Claim) The method of claim 1, wherein analyzing the code involves:

identifying loop bodies within the code; and

identifying data references to be prefetched from within the loop bodies.

6. (Original Claim) The method of claim 5, wherein analyzing the code to identify data references to be prefetched involves examining a pattern of data references over multiple loop iterations.

7. (Original Claim) The method of claim 1, wherein analyzing the code involves analyzing the code within a compiler.

8. (Original Claim) A computer-readable storage medium storing instructions that when executed by a computer cause the computer to perform a

method for generating code to perform anticipatory prefetching for data references, the method comprising:

receiving code to be executed on a computer system;

analyzing the code to identify data references to be prefetched; and

inserting prefetch instructions into the code in advance of the identified data references, wherein inserting the prefetch instructions involves,

attempting to calculate a stride value for a given data reference within a loop,

if the stride value cannot be calculated, setting the stride value to a default stride value, and

inserting a prefetch instruction to prefetch the given data reference for a subsequent loop iteration based on the stride value;

wherein the stride value is constant for some (but not necessarily all) loop iterations.

9. (Original Claim) The computer-readable storage medium of claim 8, wherein the method further comprises allowing a system user to specify the default stride value.

10. (Original Claim) The computer-readable storage medium of claim 8, wherein calculating the stride value involves:

identifying an induction variable for the stride value;

identifying a stride function for the stride value; and

calculating the stride value based upon the stride function and the induction variable.

11. (Original Claim) The computer-readable storage medium of claim 8, wherein inserting the prefetch instruction based on the stride value involves:

calculating a prefetch cover distance by dividing a cache line size by the stride value;

calculating a prefetch ahead distance as a function of a prefetch latency, the prefetch cover distance and an execution time of a loop; and

calculating a prefetch address by multiplying the stride value by the prefetch cover distance and the prefetch ahead distance and adding the result to an address accessed by the given data reference.

12. (Original Claim) The computer-readable storage medium of claim 8, wherein analyzing the code involves analyzing the code within a compiler.

13. (Original Claim) An apparatus that generates code to perform anticipatory prefetching for data references, comprising:

a receiving mechanism that is configured to receive code to be executed on a computer system;

an analysis mechanism that is configured to analyze the code to identify data references to be prefetched; and

an insertion mechanism that is configured to insert prefetch instructions into the code in advance of the identified data references;

wherein the insertion mechanism is configured to,

attempt to calculate a stride value for a given data reference within a loop,

set the stride value to a default stride value if the stride value cannot be calculated, and to

insert a prefetch instruction to prefetch the given data reference for a subsequent loop iteration based on the stride value;

wherein the stride value is constant for some (but not necessarily all) loop iterations.

14. (Original Claim) The apparatus of claim 13, further comprising a configuration mechanism that is configured to receive the default stride value from a system user.

15. (Original Claim) The apparatus of claim 13, wherein while calculating the stride value, the insertion mechanism is configured to:

identify an induction variable for the stride value;

identify a stride function for the stride value; and to

calculate the stride value based upon the stride function and the induction variable.

16. (Original Claim) The apparatus of claim 13, wherein the insertion mechanism is configured to:

calculate a prefetch cover distance by dividing a cache line size by the stride value;

calculate a prefetch ahead distance as a function of a prefetch latency, the prefetch cover distance and an execution time of a loop; and to

calculate a prefetch address by multiplying the stride value by the prefetch cover distance and the prefetch ahead distance and adding the result to an address accessed by the given data reference.

17. (Original Claim) The apparatus of claim 13, wherein the apparatus resides within a compiler.

18. (Canceled).

19. (Canceled).

20. (Canceled).

21. (Canceled).

22. (Canceled).

23. (Canceled).

24. (Canceled).

25. (Canceled).

26. (Canceled).

27. (Canceled).

28. (Canceled).

29. (Canceled).

30. (Canceled).

31. (Canceled).

32. (Canceled).

33. (Canceled).

34. (Canceled).

35. (Canceled).

36. (Canceled).

37. (Canceled).

38. (Canceled).

39. (Canceled).

40. (Canceled).

41. (Canceled).

42. (Canceled).

43. (Canceled).

44. (Canceled).

45. (Canceled).